

Рисунок 7, б - Контурне зображення.

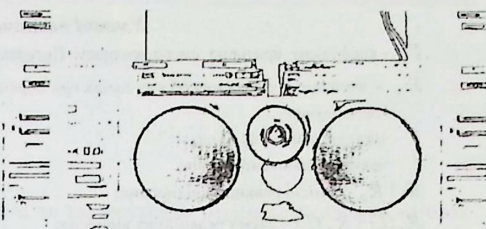


Рисунок 7, в - Контурне зображення, на яке накладено результат аналізу – товсті кола із засічками.

Три знайдені колові об'єкти показано на рис. 7, в – у вигляді товстих кіл з радіальними засічками, накладених на контурне зображення. Зауважимо, що в результаті специфіки отримання зображення (рис. 7, а) на ньому під середнім коловим об'єктом існує світловий відблиск круглої форми, якому на контурному зображенні (рис. 7, б) відповідає фрагмент у вигляді майже правильного кола. При роботі алгоритму він також розпізнався б як існуючий коловий об'єкт, проте під час застосування пропонованого методу було задано такий діапазон радіусів пошуку, який виключив таку можливість.

Нами було проведено дослідження роботи пропонованого алгоритму на ряді інших реальних зображень, в результаті чого виявлено, що при правильному налаштуванні алгоритму можна гарантувати його надійну роботу.

Література

1. Дула Р., Харт П. Распознавание образов и анализ сцен. – М.: Мир, 2000. – 510 с.
2. Прэтт У. Цифровая обработка изображений: Пер. с англ. – М.: Мир, 2002. – Кн.1, 2..

Одержано 21.12.2006 р.

УДК 681.3.07

В. Шаряк

Івано-Франківський інститут менеджменту і економіки „Галицька академія”

АРХІТЕКТУРА І КОДУВАННЯ БАЗ ДАНИХ НА ОСНОВІ ТЕОРЕТИКО-ЧИСЛОВИХ БАЗИСІВ

В статті викладено основи теоретико-числових базисів в інформаційних технологіях та їх застосування при організації бази даних. Показано перспективи використання базисів Галуа для підвищення компактності представлення інформації в базах даних. Подано приклади розрахунку ефективності рекурентних баз даних по відношенню до ієрархічних баз даних.

V. Sharyak

ARCHITECTURE AND CODING OF DATABASE ON THE BASIS OF THEORETICALLY-NUMERICAL BASES

The article is socid abaout the bases of theoretically – numerical bases in the information technologies and their use during the organization of database. The prospects of the use of Galua bases are presented to incresde the compactness of presentation of information in the given bases. The examples of calculation of efficiency of current database are set concerning the hierarchical bases of information.

Умовні позначення

- K_c - коефіцієнт критерію смердженості (інтелектуальності);
 K_c - коефіцієнт стиснення об'єму даних при переході в базис Галуа по відношенні до базису Радемахера;
 V - швидкість пошуку даних;
 L - певна множина атрибутів;
 R_1 і R_2 - сумісні реляційні відносини;
 $R_1(L)$ і $R_2(L)$ - схема реляційних відносин;
 r - кортеж відношення R ;
 m - число двійкових розрядів ідентифікатора i -го елемента БД з діапазону $1 \leq i \leq N$;
 mG - об'єм кодової інформації для ідентифікації всіх груп елементів БД;
 mR - кодова розрядність ідентифікатора в базисі Радемахера;
 mGn - ефективність кодування БД в базисі Галуа шляхом кодування числа елементів у групі після коду ідентифікатора.

Вступ

Архітектура БД – архітектура інформаційного забезпечення. Простіший варіант архітектури „клієнт/сервер (client/server)” пропонує, що СУБД повністю являє собою сервер, за винятком інтерфейсів запитів, які взаємодіють з користувачем і відсилають запити та інші команди на сервер з метою їх виконання. Сьогодні широке розповсюдження отримали багаторівневі архітектури, в яких СУБД відводиться роль постачальника динамічно генеруючого вмістимого Web-сайтів. СУБД продовжує діяти як сервер бази даних, але його клієнтом тепер служить сервер застосування (application server), який керує підключеннями до бази даних, транзакціями, авторизацією та іншими процесами. Клієнтами серверів застосування в свою чергу можуть бути Web-сервери, що обслуговують кінцевих користувачів, та інші програмні застосування [1].

Архітектура БД вперше була специфікована дослідницькою групою (ANSI/X3/SPARC Study Group on Data Base Management Systems (ANSI–Американський національний інститут стандартів, X3- його комітет обчислювальної техніки й обробки інформації, SPARC- підкомітет ANSI/X3 з планування стандартів ANSI. Метою дослідницької групи було визначення галузей і технологій баз даних, в яких було б доцільно проводити стандартизацію, а також вироблення рекомендацій для роботи в кожній із таких галузей. При цьому головним аспектом системи баз даних, який можна стандартизувати, є інтерфейси. Запропонована цією групою архітектура стала класичною та є актуальною й донині. Основною ідеєю специфікації ANSI SPARC є виділення трьох архітектурних рівнів бази даних, а саме: зовнішнього, концептуального та внутрішнього [2].

Одним із визначних факторів вдосконалення процесів зберігання та доступу до великих масивів даних є сучасні технології БД. Потужність сучасних БД базується на результатах досліджень та технологічних розробках, які отримані протягом кількох останніх десятиріч. Особливим видом баз даних є розподілені бази даних, важливими характеристиками яких є архітектурна досконалість і високий ступінь паралелізму операцій та швидкого пошуку даних. Слід зауважити, що технології розподіленої обробки інформації ускладнюють всі компоненти і змушують переглядати їх структури аж до базових елементів, причому, в багатьох розподілених БД затрати на підтримку телекомунікацій можуть переважити вартість обчислювальних процесів та програмно-технічного забезпечення [3].

Тому при дослідженні та проєктуванні розподілених БД найважливішу роль отримують задачі компактного кодування даних та швидкості обміну даними, при заданій пропускну здатності каналного зв'язку. Отже, задача вдосконалення архітектури та методів кодування інформації в розподілених БД є актуальною [3].

1 АРХІТЕКТУРНІ РІВНІ БД

Розрізняють три архітектурні рівні бази даних [2]

- 1) зовнішній;
- 2) концептуальний;
- 3) внутрішній.

1.1 Зовнішній рівень

Через зовнішній рівень користувачі та застосування отримують доступ до бази даних. Мета зовнішнього рівня – надати користувачу/застосуванню лише ті дані, які йому потрібні (а отже, до яких дозволений доступ) і в потрібному вигляді. Це індивідуальний рівень користувача, яким може бути кінцевий користувач, програміст чи застосування. Кожен з них має свою мову спілкування для кінцевого користувача – це спеціальна мова запитів, для програміста – одна з мов програмування, розширена командами звернення до СКБД, для застосування – це, як правило, певний стандартний інтерфейс звернення до бази даних через СКБД.

Зовнішня модель – це засоби зображення концептуальної моделі предметної області з урахуванням інтересів конкретних користувачів або застосувань. Кожна зовнішня модель подається в СКБД у вигляді зовнішньої схеми [2].

1.2 Концептуальний рівень

На концептуальному рівні здійснюється інтегрований опис предметної області, для якої розробляється БД, незалежно від її сприйняття окремими користувачами та способів реалізації в комп'ютерній системі. Саме через концептуальну схему здійснюється централізоване адміністрування баз даних [2].

1.3 Внутрішній рівень

Внутрішня модель є відображенням концептуальної моделі предметної області з урахуванням способів зберігання даних і методів доступу до них. Внутрішня модель відображається в СКБД у вигляді внутрішньої схеми. Внутрішня модель – це не модель фізичної пам'яті з характеристиками конкретних пристроїв зберігання даних (циліндри, доріжки тощо); воно описується у вигляді нескінченної абстрактної лінії пам'яті, яка може структуруватися за допомогою інших абстрактних понять на зразок блоків, кластерів, індексів тощо.

Доступ до фізичної пам'яті надається за допомогою опису відображень внутрішньої моделі на фізичну пам'ять операційної системи [2].

1.4 Архітектура розподілених баз даних

Розподілені бази даних (РБД) – це множина логічно взаємозалежних баз даних, розподілених у комп'ютерній мережі.

Розподілена система керування базами даних (РСКБД) – це програмне забезпечення, яке керує РБД і надає такі механізми доступу до них, що їх застосування дає користувачу можливість працювати з РБД як з однією цілісною базою даних.

Розподілені системи баз даних (РСБД) – це РБД разом із РСКБД [2,3].

Архітектура розподілених баз даних представлена на рис.1.



Рисунок 1- Архітектура розподілених баз даних.

2 ХАРАКТЕРИСТИКИ РІЗНИХ ТИПІВ БД

Розрізняють п'ять типів моделей баз даних [4]:

- | | |
|----------------|--------------------------|
| 1) ієрархічні; | 4) лінійно-рекурентні; |
| 2) мережеві; | 5) об'єктно-орієнтовані; |
| 3) реляційні; | 6) об'єктно-реляційні. |

2.1 Ієрархічні бази даних

Ієрархічна структура даних - це структура, де будь-який об'єкт може підпорядковуватися лише одному об'єкту вищого рівня, а йому багато об'єктів нижчого рівня. Такий зв'язок між даними називають „один до багатьох”. За ієрархічним принципом побудовані файлові структури даних на дисках персонального комп'ютера [4].

2.2 Мережева база даних

Мережева структура характеризується тим, що будь-який об'єкт одного рівня (одної групи даних) може мати довільні зв'язки з об'єктами іншого рівня. Такі зв'язки називаються „багато до багатьох”. Мережеві структури можна описати у вигляді таблиці, де у першому горизонтальному рядку записують об'єкти одного рівня, а у першому вертикальному – іншого. Така таблиця добре ілюструє зв'язки між об'єктами, але може мати багато порожніх елементів, що призводить до значної надлишковості БД [4].

2.3 Реляційні бази даних

Реляційна БД є найпоширенішим типом і характеризується поданням даних у вигляді декількох таблиць і зв'язаними між собою кортежів та атрибутів. Один з найпростіших типів зв'язків є „один до одного”. Потрібні дані будуть черпатися з двох таблиць. Отже, для зв'язку між таблицями використовують поле, значення якого не повторюється в різних записах. Це поле називається ключовим. Якщо реляційні таблиці мають спільні поля, то зміни в спільному полі в одній таблиці автоматично відображаються у всіх таблицях. Мета запровадження реляційних зв'язків – мінімізувати дублювання даних і забезпечити можливість опрацювати (шукати) дані з декількох таблиць, що забезпечує значне зменшення надлишковості БД [4].

2.3 Лінійно-рекурентні бази даних

Лінійно-рекурентні БД є новим типом структурної організації баз даних, яка формується на основі рекурентних властивостей теоретико-числового базису Галуа. Важливою характеристикою таких БД є максимальна компактність адресації даних, в тому числі при багаторівневій архітектурі. Позитивною характеристикою лінійно-рекурентних БД є можливість ефективного захисту від помилок та несанкціонованого доступу [4].

2.4 Об'єктно-орієнтовані бази даних (Object Definition Language) ODL

Один із підходів до забезпечення сумісності систем баз даних з парадигмою об'єктно-орієнтованого проектування зв'язаний з розширенням системи розуміння, що в основі є об'єктно-орієнтовані мови програмування, такі як C++ або Java, в напрямку підтримки засобів постійного зберігання даних. В традиційному програмуванні розуміється, що після завершення циклу роботи програми її об'єкти безповоротно втрачаються, в той час як принципові умови роботи будь-якої СУБД полягають у тому, що об'єкти повинні зберігатися не обмежено довгий час, поки не будуть змінені або знищені примусово, як це відбувається у файлових системах. „Чисту” об'єктно-орієнтовану модель представлення даних, названу *ODL (Object Definition Language - мова визначення об'єктів)*, в свій час було стандартизовано дослідницькою групою Object Data Management Group (ODMG) [4].

2.5 Об'єктно-реляційні бази даних (Object-relational model)

Ця модель є частиною останнього стандарту SQL, названого SQL-99 (або SQL-1999, також SQL3), і є варіантом розширення звичайної реляційної моделі за рахунок формалізації багатьох загальноприйнятих концепцій об'єктно-орієнтованого проєктування. Вказаний стандарт служить основою для побудови об'єктно-реляційних систем баз даних, які сьогодні випускаються всіма основними постачальниками комерційних СУБД. Ці системи, однак, значно відрізняються в деталях практичної реалізації початкових концепцій [4].

2.6 Напівструктуровані бази даних (Semistructured data)

Одним із останніх досягнень, призначених вирішити велику кількість актуальних проблем технологій СУБД, включаючи необхідність об'єднання традиційних баз даних з іншими джерелами інформації, такими, як Web-сторінки найрізноманітнішої структури, є розробка технологій організації напівструктурованих БД. В цей час як об'єктно-орієнтовані або об'єктно-реляційні системи пропонують використання фіксованих схем для кожного класу або відношень, модель напівструктурованих даних відрізняється набагато більш високою гнучкістю представлення компонентів інформації. Найбільш практичним втіленням моделі напівструктурованих даних є XML (eXtensible Markup Language - розширювана мова розмітки). XML - це специфікація опису „документів”, що представляють собою набір вкладених елементів даних, функції яких визначаються відповідними тегами. Підтримка XML найважливішим компонентом систем, що здійснюють посередницькі функції між різнорідними джерелами даних, реалізує цілі гнучкого представлення інформації в базах даних [4].

2.7 Методи оптимізації та підвищення ефективності БД

Основними характеристиками оптимізації і підвищення ефективності є :

- здешевлення системи СУБД ;
- збільшення швидкості пошуку доступу до даних;
- зменшення або стиснення об'єму інформації ;
- розробка різноманітних типів третинних пристроїв зберігання (tertiary storage devises) даних;
- тенденції росту системи ;
- підвищення продуктивності системи ;
- паралельні обчислення;
- розвиток багаторівневої архітектури;
- підвищення надійності захисту від помилок;
- захист від несанкціонованого доступу до БД [1,2].

2.8 Критерії ефективності БД

Важливими характеристиками ефективності БД є:

- емерджентність [1,2]

$$K_e = \frac{N_3}{N_0}, \quad (1)$$

де N_3 - число зв'язків, N_0 - число елементів ($K_e \geq 2$);

- надлишковість

$$K_e = \frac{mR}{mG}, \quad (2)$$

де $\frac{mR}{mG}$ - відповідні об'єми кодової інформації в базисі Радсмахера та інших базисах;

- швидкість пошуку даних

$$V = \frac{\Delta N_i}{T}, \quad (3)$$

де ΔN_i - об'єм шуканих i -х даних в БД; T - час доступу зчитування ΔN_i даних;

- формалізація перетворення даних [2];

- об'єднання

$$R(L) = R_1(L) \cup R_2(L) = \{r \mid r \in R_1 \vee r \in R_2\}, \quad (4)$$

де L - певна множина атрибутів; R_1, R_2 - об'єднання сумісних реляційних відношень;

- перетин

$$R(L) = R_1(L) \cap R_2(L) = \{r \mid r \in R_1 \& r \in R_2\}; \quad (5)$$

- різниця

$$R(L) = R_1(L) - R_2(L) = \{r \mid r \in R_1 \& r \notin R_2\}; \quad (6)$$

- проекція

$$S = R[A_1, \dots, A_m] = \{r[A_1, \dots, A_m] \mid r \in R\}, \quad (7)$$

де R - проекція реляційного відношення зі схемою $R(A_1, \dots, A_k)$ за атрибутами $A_1, \dots, A_m$, де $\{A_1, \dots, A_m\} \subset \{A_1, \dots, A_k\}$;

- обмеження

$$S = R[L \theta M] = \{r \mid r \in R \& r[L] \theta r[M]\}, \quad (8)$$

де θ - порівняння атрибутів, L і M - набори θ - порівнювальних атрибутів схеми відношення R ;

- декартів добуток

$$Q = R \times S = \{(r, s) \mid r \in R \& s \in S\}; \quad (9)$$

- з'єднання

$$Q = R[L \theta M]S = \{(r.s) \mid r \in R \& s \in S \& r[M] \theta s[N]\}, \quad (10)$$

де відношення R має схему $R(L, M)$, а відношення S - схему $S(M, N)$, M і N - множини атрибутів, θ - порівняння;

- ділення

$$R[N \div K]S = R[M] - ((R[M] \times S[K]) - R[M]), \quad (11)$$

де відношення R і S зі схемами $R(M, N)$ та $S(K, L)$, для яких проекції $R[N]$ та $S[K]$ є сумісними.

3 ЗАСТОСУВАННЯ ТЕОРЕТИКО-ЧИСЛОВИХ БАЗИСІВ ПРИ ОРГАНІЗАЦІЇ БАЗ ДАНИХ

Теоретико-числові базиси - є фундаментальними теоретичними основами систем числення та методів кодування даних. (ТЧБ) - породжуються системами ортогональних функцій (наприклад: базиси, функції Фур'є, поліноми Лагранжа, Чебишева, Бесселя, функції Якобі, та інші) [5].

В даний час в теорії та техніці цифрової обробки даних спостерігається ефективне використання теоретико-числових базисів (ТЧБ). Аналіз використання теоретико-числових базисів в інформаційних технологіях показує виключно широке застосування базису Радемахера, який має ще більш компактну кодову матрицю і породжує двійкові кодові системи числення. В базисі Радемахера працюють всі сучасні комп'ютерні системи [5].

Сьогодні світовий досвід застосування ТЧБ для швидкодіючих теоретико-числових перетворень, які реалізують числові згортки та цифрову обробку сигналів, показує що значне прискорення швидкості процесорів та упрощення програмних алгоритмів забезпечують базиси Крестенсона та Галуа [5].

Очевидно, що застосування базисів з високою надлишковістю кодових матриць, наприклад, Унітарний, Хаара, Крейга, є несефективними при кодуванні інформації у базах даних. Тому аналіз ефективності застосування ТЧБ в базах даних виконаємо на основі базисів Радемахера, Крестенсона і Галуа [5].

В основу ТЧБ Крестенсона покладена система пилоподібних трикутних ортогональних функцій. Базис Крестенсона породжує непозиційну систему числення залишкових класів (СЗК) [5].

Базис Галуа є найбільш компактним з відомих ТЧБ. Кодування даних в базисі Галуа є рекурентним, і описується теорією полів Галуа. Таким чином ТЧБ Галуа має кодову матрицю, яка перетворюється у вектор [5].

3.1 Порівняльна характеристика матриці Радемахера з вектором Галуа

Базис Галуа породжує систему числення Галуа. Наприклад, при $m=4$: $N=2^4=16$ кодовий вектор базису Галуа описується двійковою послідовністю 11110101110010000 (табл.1).

Таблиця 1 - Порівняльна характеристики матриці Радемахера з вектором Галуа

Матриця Радемахера			Вектор Галуа	
0 0 0 0	0		0	
0 0 0 1	1	→	0	
0 0 1 0	2	→	0	
0 0 1 1	3		0	
0 1 0 0	4		1	
0 1 0 1	5		0	
0 1 1 0	6	→	0	
0 1 1 1	7		1	
1 0 0 0	8		1	
1 0 0 1	9		0	
1 0 1 0	10		1	
1 0 1 1	11	→	0	
1 1 0 0	12		1	
1 1 0 1	13		1	
1 1 1 0	14		1	
1 1 1 1	15		1	

З табл. 1 видно, що для кодування чисел в базисі Радемахера в діапазоні 0-N необхідно $\log_2 N$ розрядів двійкових чисел. При кодуванні в базисі Галуа на кожне N , число припадає 1 біт Галуа. А для кодування чисел N з вектора необхідно рекурентно вибрати m -біт даних, що засвідчує максимальну упаковку інформації в базисі Галуа.

4 ЛІНІЙНО-РЕКУРЕНТНА АРХІТЕКТУРА В БАЗИСІ ГАЛУА

При застосуванні базису Радемахера реалізується метод кодування елементів ієрархічних БД, що представлено архітектурою на рис.2 [5].

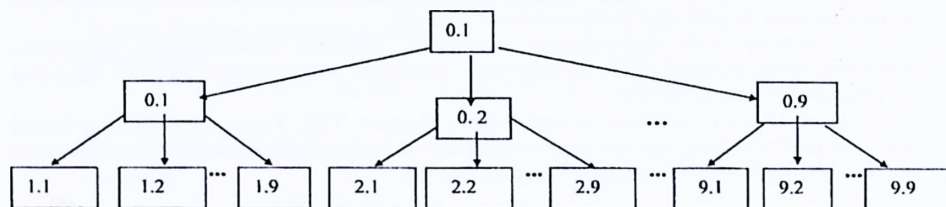


Рисунок 2 - Ієрархічна БД в базисі Радемахера.

При цьому для кодування елементів ієрархічної БД у базисі Радемахера або Крестенсона необхідно дані кодувати згідно з кодовою матрицею (табл.1).

Очевидно, що даній кодовій матриці базису Радемахера однозначно відповідає матриця-вектор базису Галуа (табл.1).

Ієрархічна архітектура БД в базисі Галуа може бути представлена новою лінійно - рекурентною архітектурою (рис. 3).

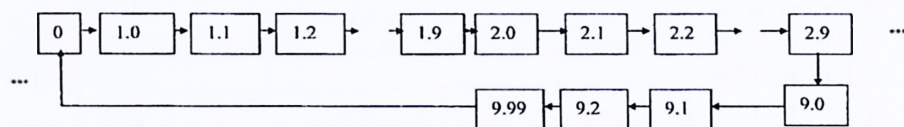


Рисунок 3 - Лінійно - рекурентна архітектура БД в базисі Галуа.

Властивість рекурентності зв'язків ієрархічної БД при кодуванні елементів її в базисі Галуа дозволяє повному розглянути характеристики організації таких БД.

Для БД, показаних на рис.2, рис.3, визначасмо коефіцієнт стиснення

$$K_{\epsilon} = \frac{mR}{mG} = \frac{800}{100} = 8, \quad (12)$$

де $mR = 100 \cdot 8 \text{ біт} = 800 \text{ біт}$, $mG = 100 \text{ біт}$.

5 БАГАТОРІВНЕВІ БАЗИ ДАНИХ У БАЗИСІ ГАЛУА

На основі лінійно-рекурентної архітектури БД у базисі Галуа може бути побудована багаторівнева структура БД (рис.4).

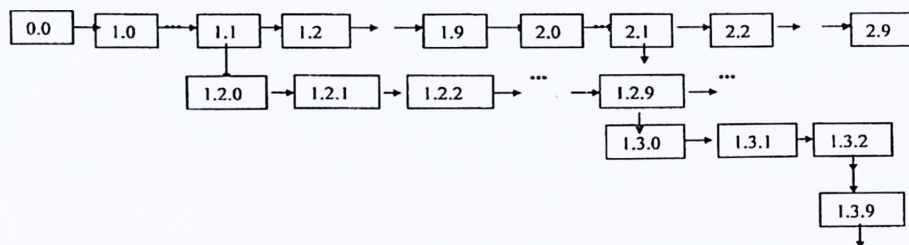


Рисунок 4 - Багаторівнева архітектура БД в базисі Галуа.

6 ПОРІВНЯННЯ ЕФЕКТИВНОСТІ КОДУВАННЯ ЕЛЕМЕНТІВ БД У РІЗНИХ БАЗИСАХ

Нехай маємо ієрархічну БД з бінарним кодуванням елементів на різних рівнях:
 $N = 5000$, $n = 4$. Необхідно виконати адресацію 256 елементів, згрупованих в три групи: 200, 50, 6.

Виходячи з максимального числа елементів БД визначимо розрядність коду ідентифікатора елементів у базисі Галуа

$$m = \sum_{i=1}^n [\log_2 5000] = 13 \text{ біт.} \quad (13)$$

Таким чином, об'єм кодової інформації для ідентифікації всіх груп елементів БД описується наступним чином

$$mG = 3 \cdot 13 + 200 + 50 + 6 = 296 \text{ біт.}$$

Аналогічно кодова розрядність ідентифікатора в базисі Радемахера визначається наступним чином

$$mR = 13 \cdot 256 = 3328 \text{ біт,}$$

звідки коефіцієнт стиснення ідентифікатора елементів БД в базисі Галуа

$$K_c = \frac{mR}{mG} = \frac{3328}{296} = 11,24. \quad (14)$$

Якщо в БД елементи тісно пов'язані між собою і не перевищують $n=100$, то ефективність кодування БД в базисі Галуа можна підвищити шляхом кодування числа елементів у групі після коду ідентифікатора, тобто:

$$mGn = 3 (13 + \sum_{i=1}^n [\log_2 100]) = 3 (13 + 7) = 60 \text{ біт.} \quad (15)$$

Звідси коефіцієнт стиснення об'єму даних багаторівневої БД при переході в базис Галуа по відношенні до базису Радемахера дорівнює:

$$K_c = \frac{mR}{mGn} = \frac{3328}{60} = 55,46. \quad (16)$$

Висновки

Проведені дослідження архітектури БД показують, що ієрархічні, мережеві та реляційні архітектури БД не задовільняють умови смисловості [4].

Тому перспективним є розвиток та вдосконалення архітектури БД об'єктно-орієнтованих та об'єктно-реляційних архітектур.

Дослідження теоретико-числових базисів та їх застосування при організації БД показують, що традиційний спосіб ідентифікації та кодування елементів БД в базисі Радемахера є неефективним, надлишковим і завадонебезпечним, крім того, традиційний метод є незахищеним від несанкціонованого доступу. В той же час дані характеристики БД достатньо ефективно реалізуються в базисі Галуа.

Очевидно, що глибокі дослідження застосування теорії і методів кодування даних та організації БД в базисі Галуа є актуальними і перспективними.

Запропоновані багаторівневі структури БД в базисі Галуа можуть бути ефективно застосовані для компактного завадозахищеного (таємного) кодування даних, наприклад, в багаторівневному маркстинзі (БРМ) біоенергетичних активних продуктів п'ятого покоління (БАД) тибетської і китайської народної медицини в міжнародній китайській транснаціональній корпорації „ТЯНЬ ШІ”.

Література

1. Дейт К. Введение в системы баз данных. -М.: дом „Вильямс“, 2005. -1328 с.
2. Пасічник В.В., Резніченко В.А. Організація баз даних та знань. -К.: Видавнича група БНВ, 2006. - 384 с.
3. Диго С.М. Проектирование и использование баз данных. -М.: Финансы и статистика. 1995.-346с.
4. Шаряк В.В., Николайчук Я.М. Системні характеристики баз даних та перспективні напрямки їх розвитку. -Київ.: Науковий журнал інституту електродинаміки НАН України. 2006. -С. -7-12.
5. Шаряк В.В., Николайчук Я.М. Теоретико-числові базиси да їх застосування при організації баз даних. - Вінниця.: Міжнародний науково-технічний журнал „Оптико-електронної інформаційно-енергетичні технології”. 2006. -С.10-16.

Одержано 26.12.2006 р.